LKP 9

Danang Wahyu N (G64150003)
Yudha Prasetya S (G64150038)
Devy Apriansyah (G64150044)
Octavian Wibawa R (G64150087)
Noer Widya Herlambang (G64150105)

Langkah Kerja

1. Masukan seluruh pustaka yang diperlukan

```
#import
library(qdap)
library(lsa)
library(NMF)
```

2. Baca fail text dan stopword yang dipakai. Hilangkan kolom class pada cluster.csv

```
#read fail csv
doc_csv = read.csv("cluster.csv")
keeps = c("ID", "ABSTRACT")
doc = doc_csv[keeps]
#read fail stopword
stopword = scan("stopwords.txt", character(), quote = "", sep = ",")
```

3. Bangkitkan matrix Term Frequency dan hapus stopword pada matrix itu. Ide nya adalah menghapus kolom pada matrix Term Frequency yang merupakan stop word.

```
#hapus stop words
tf.matrix = t(with(doc, wfm(ABSTRACT, ID)))
tf.matrix = as.data.frame(tf.matrix)
tf.matrix.clean = tf.matrix[ , !(names(tf.matrix) %in% stopword)]
tf.matrix.clean = t(tf.matrix.clean)
#tf.matrix.clean.example = tf.matrix.clean [1:5, ]
View(head(tf.matrix.clean))
```

4. Bangkitkan matrix cosine dissimilarity. Kami menggunakan dissimilarity karena fungsi hclust() yang dipakai menerima masukan dissimilarity. Caranya adalah membuat matrix similarity kemudian dilakukan fungsi pengurangan dengan matrix yang berelemen 1.

```
#clustering
###Cosine similarity
coss <- function(x) {crossprod(x)/(sqrt(tcrossprod(colSums(x^2))))}
C = coss(as.matrix((tf.matrix.clean)))
###Cosine Disimilarity
B = matrix(1, nrow = nrow(C), ncol = ncol(C))
cos.dis = B - C
```

5. Lakukan Proses Cluster dan perhitungan purity. Dibuat tujuh buah cluster karena jumlah class data masukan memiliki tujuh class.

```
## Ave distance
hc = hclust(as.dist(cos.dis), "ave")
cluster = cutree(hc, k = 7)
class = as.vector(doc_csv[2])
plot(hc, main = "Clustering average distance")
rect.hclust(hc, 7, border = "red")
purity(class, cluster)

## single distance
hc = hclust(as.dist(cos.dis), "single")
cluster = cutree(hc, k = 7)
class = as.vector(doc_csv[2])
plot(hc, main = "Clustering single distance")
rect.hclust(hc, 7, border = "red")
purity(class, cluster)

## complete distance
hc = hclust(as.dist(cos.dis), "complete")
cluster = cutree(hc, k = 7)
class = as.vector(doc_csv[2])
plot(hc, main = "Clustering complete distance")
rect.hclust(hc, 7, border = "red")
purity(class, cluster)
```

Hasil Purity

Kesimpulan